UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/845,751	04/30/2001	Bernhard J. Scholz	GE1-004US	3458

21718   7590   08/08/2005

LEE & HAYES PLLC
SUITE 500
421 W RIVERSIDE
SPOKANE, WA 99201

EXAMINER
PAULA, CESAR B

ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 08/08/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _01 June 2005_.

2a)☒ This action is **FINAL.**   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle,* 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-6 and 9-43_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-6 and 9-43_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to the response to the amendment filed on 6/1/2005.

This action is made Final.

2.   In the amendment, claim 43 has been added. Claims 1-6, and 19-43 are pending in the

case. Claims 1, 19, 23, 32, and 38 are independent claims.

Drawings

3.   The drawings filed on 4/30/2001 have been approved by the Examiner.

Claim Rejections - 35 USC § 112

4.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the
> subject matter which the applicant regards as his invention.

5.   The 35 U.S.C. 112, second paragraph rejection of claim 28 has been withdrawn as

necessitated by the amendment.

6.   Claim 4 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing

to particularly point out and distinctly claim the subject matter which applicant regards as the

invention.

7.   Claim 4 recites the limitation "the field" in line 2. There are two fields in claim 1. It is

unclear which field is being referred in this claim. There is insufficient antecedent basis for this

limitation in the claim.

Claim Rejections - 35 USC § 102

8.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed
in the United States before the invention by the applicant for patent or (2) a patent granted on an application for
patent by another filed in the United States before the invention by the applicant for patent, except that an
international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this
subsection of an application filed in the United States only if the international application designated the United
States and was published under Article 21(2) of such treaty in the English language.

9.   Claims 1, and 19 remain, and 43 is rejected under 35 U.S.C. 102(e) as being anticipated

by Raz (Pat.# 6,292,827, 9/18/2001, filed on 6/20/1997).

Regarding independent claim 1, Raz teaches the automatic conversion of paper forms

into HTML-coded forms having form fields—*custom field on a source code form definition.*

During the automatic conversion, validation functions are added to the converted HTML-coded

form fields for verifying the data input into those fields—*identifying a custom field on a HTML*

source code form definition and one or more restrictions and validation code, that when

executed, validates that the input conforms to the one or more restrictions (col.12, lines 36-49).

In other words, a new HTML file is formed, which comprises not only the coded fields, but also

the validation functions associated with corresponding fields-- *adding to a new form definition*

that includes a non-custom field corresponding to the custom field, the identified validation

code..

Regarding independent claim 19, Raz teaches the automatic conversion of paper forms

into HTML forms. During the automatic conversion, validation functions are added to the

converted HTML-coded form fields for verifying the data input into those fields—*identifying,*

from an input source code definition written in a source code, one or more desired fields and

automatically adding validation code to source code of the form (col.12, lines 36-49).

Regarding claim 43, which depends on claim 1, Raz teaches the adding of validation

functions to HTML-coded form fields. For example, a digit-checking validation function is

added to an account number field found in the HTML form—*using, in identifying the validation*

code, the one or more restrictions (col.12, lines 36-49). In other words, the form field is

evaluated to determine the type of data it will receive, in this case account digits, then a digit-

checking validation form is retrieved and added to the HTML form.

Claim Rejections - 35 USC § 103

10.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art
> are such that the subject matter as a whole would have been obvious at the time the invention was made to
> a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be
> negatived by the manner in which the invention was made.

11.   Claims 2-6, 20-24, and 26-42 remain rejected under 35 U.S.C. 103(a) as being

unpatentable over Raz, in view of Laura Lemay's Workshop JavaScript, Lemay et al, hereinafter

Lemay, 1996, Sams.net, pp.132-137.

Regarding claim 2, which depends on claim 1, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions—*validation*

code that causes the other processor to execute the identified validation code --are added to the

HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to

explicitly disclose: *adding to the new form definition, a reference to the identified validation*

code. However, Lemay teaches adding validation functions to HTML forms using a function

call, such as "function Validate ()" page 134, line 3-- *reference to the added validation code.* The

validation functions are added within HTML tags (page 133, lines 17-page 18). It would have

been obvious to a person of ordinary skill in the art at the time of the invention to add the

validation reference to the converted form of Raz with well-known HTML function calls taught

by Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback

without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and

trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding claim 3, which depends on claim 1, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions—*pre-defined*

validation code are added to the HTML form fields for verifying the data input into those fields

(col.12, lines 36-49).

Regarding claim 4, which depends on claim 1, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions are added to the

HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to

explicitly disclose: *the source code form definition that defines a field includes a tag*

corresponding to the field. However, Lemay teaches adding validation functions to HTML

forms. The validation functions are added within HTML tags (page 133, lines 17-page 18). It

would have been obvious to a person of ordinary skill in the art at the time of the invention to

add the validation reference to the converted form of Raz with well-known HTML function calls

taught by Lemay, because Lemay teaches the saving of trouble, and receiving immediate

feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save

time and trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding claim 5, which depends on claim 1, Raz teaches the automatic conversion of

paper forms into HTML forms, having fields, such as an account field, for typing—*user input--*

an account number (col.12, lines 36-49).

Regarding claim 6, which depends on claim 1, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input into those fields—*identifying, from a*

plurality of pieces of validation code, the validation code corresponding to the one or more

attributes of the custom tag (col.12, lines 36-49). Raz fails to explicitly disclose: *identifying a*

custom tag corresponding to the field, wherein the custom tag includes an indication of one or more attributes and wherein each of the one or more attributes includes a value indicating what input corresponding to the non-custom field is to be restricted to. However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags, and the functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to include the validation functions of Raz within tags of Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags—*custom tags.*

Regarding claim 20, which depends on claim 19, Raz teaches the automatic conversion of paper forms into HTML forms. During the automatic conversion, validation functions are added to the converted form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to explicitly disclose: *a custom tag corresponding to each of the one or more desired fields, wherein each custom tag has on e or more validation attributes and wherein each validation attribute includes an indication of the attribute and a corresponding value that input corresponding to the custom tag is to be restricted to.* However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags, and the functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to include the validation functions of Raz within tags of Lemay,

because Lemay teaches the saving of trouble, and receiving immediate feedback without having

to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by

inserting the Javascript validation functions in the created HTML tags—*custom tags*.

Regarding claim 21, which depends on claim 19, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input by a user into those fields (col.12, lines

36-49).

Regarding claim 22, which depends on claim 19, Raz teaches the automatic conversion of

paper forms into HTML forms—*generating a temporary form definition*. During the automatic

conversion, validation functions are added to the converted form fields for verifying the data

input into those fields–*adding and executing code to add the identified validation code to the*

new form definition and outputting, as the source code, the temporary form definition (col.12,

lines 36-49).

Regarding independent claim 23, Raz teaches the automatic conversion of paper forms

into HTML forms. After converting the forms, validation functions are added to the HTML form

fields for verifying the data input into those fields — *further to add to a form definition, for each*

of the one or more custom tags, validation code to validate subsequent inputs to a field (col.12,

lines 36-49). Raz fails to explicitly disclose: *one or more custom tags in a source code form*

definition; and replace each of the one or more custom tags with another tag, and field

corresponding to the tag. However, Lemay teaches adding validation functions to HTML forms.

The validation functions are added within HTML tags in a form file, and the functions indicate

an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-

page 18). It would have been obvious to a person of ordinary skill in the art at the time of the

invention to include the validation functions into the HTML form of Raz within tags as

suggested by Lemay, because Lemay teaches the saving of trouble, and receiving immediate

feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save

time and trouble by inserting the Javascript validation functions in the created HTML tags—

custom tags.

Regarding claim 24, which depends on claim 23, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input by a user into those fields (col.12, lines

36-49).

Regarding claim 26, which depends on claim 23, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions are added to the

HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to

explicitly disclose: *a custom tag is a HTML tag.* However, Lemay teaches adding validation

functions to HTML forms. The validation functions are added within HTML tags (page 133,

lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time

of the invention to replace the tags of the converted form of Raz with well-known HTML tags

suggested by Lemay, because Lemay teaches the saving of trouble, and receiving immediate

feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save

time and trouble by inserting the Javascript validation functions in the created HTML tags—

custom tags.

Regarding claim 27, which depends on claim 23, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions are added to the

HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to

explicitly disclose: *add a reference to the added validation code*. However, Lemay teaches

adding validation functions to HTML forms using a function call, such as "function Validate ()"

page 134, line 3-- *reference to the added validation code*. The validation functions are added

within HTML tags (page 133, lines 17-page 18). It would have been obvious to a person of

ordinary skill in the art at the time of the invention to add the validation reference to the

converted form of Raz with well-known HTML function calls taught by Lemay, because Lemay

teaches the saving of trouble, and receiving immediate feedback without having to wait on a

server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the

Javascript validation functions in the created HTML tags.

Regarding claim 28, which depends on claim 23, Raz teaches the automatic conversion of

paper forms into HTML forms—*form definition*. After converting the forms, validation functions

are added to the HTML form fields-- *generate a new document corresponding to the form*

definition-- for verifying the data input into those fields-- *add validation code to the new*

document (col.12, lines 36-49). Raz fails to explicitly disclose: *to replace each of the one or*

more custom tags with another tag by adding the other tag to the new document. However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to add the validation code to the newly converted form of Raz within well-known HTML tags taught by Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding claim 29, which depends on claim 23, Raz teaches the automatic conversion of paper forms into HTML forms. After converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields—*restrictions corresponding to the same validation code, add the same validation code only once* (col.12, lines 36-49). Raz fails to explicitly disclose: *one or more custom tags.* However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to add the validation code to the newly converted form of Raz within well-known HTML tags taught by Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding claim 30, which depends on claim 23, Raz teaches the automatic conversion of paper forms into HTML forms. After converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields. The data, including validation functions, are stored in a RDBMS database—*tag library* (col. 9, lines 27-45, col.12, lines 36-49).

Regarding claim 31, which depends on claim 30, Raz teaches the automatic conversion of paper forms into HTML forms. After converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields. The data, including validation functions, are stored in a RDBMS database—*tag library* (col. 9, lines 27-45, col.12, lines 36-49). Raz fails to explicitly disclose: *an identification of the one or more custom tags.* However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags--*identification* (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to combine Raz, and Lemay to store the well-known HTML tags taught by Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding independent claim 32, Raz teaches the automatic conversion of paper forms into HTML forms—*form definition*(col.12, lines 36-49).

Moreover, Raz teaches that after converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields — *identifying and adding*

validation code that, when executed based on an input corresponding to the field, *validates whether the associated restrictions are satisfied* (col.12, lines 36-49). In other words, the converted HTML-coded form(s) is received, and then validation functions associated—*one or more associated input restrictions*-- with the HTML form fields, are added to the form. Raz fails to explicitly disclose: *identifying a replacement non-custom tag, adding the identified replacement non-custom tag to a new form definition.* However, Lemay teaches adding validation functions to HTML forms along with tags— *identifying and adding a replacement non-custom tag.* The validation functions are added within HTML tags, and the functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to have added the validation tags of Lemay and functions to the form fields of Raz, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags—*custom tags.*

Regarding claim 33, which depends on claim 32, Raz teaches the automatic conversion of paper forms into HTML forms. After converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to explicitly disclose: *adding to the new form definition, a reference to invoke the added validation code.* However, Lemay teaches adding validation functions to HTML forms using a function call, such as "function Validate ()" page 134, line 3-- *reference to the added validation code.* The validation functions are added within HTML tags (page 133, lines 17-page 18). It would have

been obvious to a person of ordinary skill in the art at the time of the invention to add the

validation reference to the converted form of Raz with well-known HTML function calls taught

by Lemay, because Lemay teaches the saving of trouble, and receiving immediate feedback

without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and

trouble by inserting the Javascript validation functions in the created HTML tags.

Regarding claim 34, which depends on claim 32, Raz teaches that after converting the

forms—*receiving* HTML *form definition*, validation functions are added to the HTML form

fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to explicitly

disclose: *adding the each of the non-custom tags to the new form definition.* However, Lemay

teaches adding validation functions to HTML forms along with tags— *identifying and adding a*

replacement non-custom tag. The validation functions are added within HTML tags, and the

functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page

133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the

time of the invention to have added the validation tags of Lemay and functions to the form fields

of Raz, because Lemay teaches the saving of trouble, and receiving immediate feedback without

having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble

by inserting the Javascript validation functions in the created HTML tags—*custom tags*.

Regarding claim 35, which depends on claim 32, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input by a user into those fields (col.12, lines

36-49).

Regarding claim 36, which depends on claim 32, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input into those fields (col.12, lines 36-49).

Raz fails to explicitly disclose: *each input custom tag includes one or more attributes that*

identify the one or more associated restrictions, and wherein each of the one or more attributes

includes an indication of the attribute and a corresponding value for that data input

corresponding to the tag is to be restricted to. However, Lemay teaches adding validation

functions to HTML forms. The validation functions are added within HTML tags, and the

functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page

133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the

time of the invention to include the validation functions of Raz within attributes of Lemay,

because Lemay teaches the saving of trouble, and receiving immediate feedback without having

to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by

inserting the Javascript validation functions in the created HTML tags.

Regarding claim 37, which depends on claim 32, Raz teaches the automatic conversion of

paper forms into HTML forms. During the automatic conversion, validation functions are added

to the converted form fields for verifying the data input into those fields–*execution code to add*

the identified validation code to the new form definition (col.12, lines 36-49).

Regarding independent claim 38, Raz teaches the automatic conversion of paper forms into HTML forms, having fields for entering data. After converting the forms, validation functions are added to the HTML form fields—*a first portion identifying an input field--* for verifying the data input into those fields — *a second portion identifying validation code to be added to a page to enforce the one or more restrictions* (col.12, lines 36-49).

Regarding claim 39, which depends on claim 30, Raz teaches the automatic conversion of paper forms into HTML forms—*text markup language document.* After converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields (col.12, lines 36-49).

Regarding claim 40, which depends on claim 38, Raz teaches the automatic conversion of paper forms into HTML forms, having fields, such as an account field, for typing an account number (col.12, lines 36-49).

Regarding claim 41, which depends on claim 38, Raz teaches the automatic conversion of paper forms into HTML forms. During the automatic conversion, validation functions are added to the converted form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to explicitly disclose: *one or more attributes and for each attribute and associated value for the attribute.* However, Lemay teaches adding validation functions to HTML forms. The validation functions are added within HTML tags, and the functions indicate an attribute,

such as "ValidLength", and length value not to be exceeded (page 133, lines 17-page 18). It

would have been obvious to a person of ordinary skill in the art at the time of the invention to

include the validation functions of Raz within attributes of Lemay, because Lemay teaches the

saving of trouble, and receiving immediate feedback without having to wait on a server (page

132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript

validation functions in the created HTML tags.

Regarding claim 42, which depends on claim 38, Raz teaches the automatic conversion of

paper forms into HTML forms, having fields, such as an account field, for typing—*user input--*

an account number (col.12, lines 36-49).

12.   Claim 25 remains rejected under 35 U.S.C. 103(a) as being unpatentable over Raz in

view of Lemay, and further in view of Kryka et al, hereinafter Kryka (US Pat.# 6,832,369 B1,

12/14/2004, filed on 8/1/2000).

Regarding claim 25, which depends on claim 23, Raz teaches the automatic conversion of

paper forms into HTML forms. After converting the forms, validation functions are added to the

HTML form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to

explicitly disclose: *the system comprises a compiler.* However, Kryka teaches a Java compiler

for compiling source code into Java bytecode form (col.1, lines 54-68). It would have been

obvious to a person of ordinary skill in the art at the time of the invention to have included a

compiler, because Kryka teaches above the popular use of a platform-independent programming

language--Java. Thus, allowing a user to easily implement the form across multiple computer platforms.

Response to Arguments

13.   Applicant's arguments filed 6/1/2005 have been fully considered but they are not persuasive. The Applicants indicate that Raz does not teach or suggest the amended limitations of identifying a custom field on a source code form definition and one or more restrictions on an input to the custom field (pages 9-10). The Examiner disagrees, because Raz teaches the automatic conversion of paper forms into HTML-coded forms having form fields—*custom field on a source code form definition*. During the automatic conversion, validation functions are added to the converted HTML-coded form fields for verifying the data input into those fields— *identifying a custom field on a* HTML *source code form definition and one or more restrictions and validation code, that when executed, validates that the input conforms to the one or more restrictions* (col.12, lines 36-49). In other words, a new HTML file is formed, which comprises not only the coded fields, but also the validation functions associated with corresponding fields-- *adding to a new form definition that includes a non-custom field corresponding to the custom field, the identified validation code*.

        Regarding claims 2-6, the Applicants submit that Lemay does not cure the deficiencies of Raz, and therefore the claims are allowable based on their dependency on claim 1 (page 11). As discussed above the limitations of claim 1 as amended are taught by Raz.

Regarding claim 23, the Applicants indicate that Raz does not teach or suggest the

amended limitations of the identification of custom tags on a source code form definition (page

12). The Examiner disagrees, because Raz teaches the automatic conversion of paper forms into

HTML forms. After converting the forms, validation functions are added to the HTML form

fields for verifying the data input into those fields —*further to add to a form definition, for each*

of the one or more custom tags, validation code to validate subsequent inputs to a field (col.12,

lines 36-49). Raz fails to explicitly disclose: *one or more custom tags in a source code form*

definition; and replace each of the one or more custom tags with another tag, and field

corresponding to the tag. However, Lemay teaches adding validation functions to HTML forms.

The validation functions are added within HTML tags in a form file, and the functions indicate

an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-

page 18). It would have been obvious to a person of ordinary skill in the art at the time of the

invention to include the validation functions into the HTML form of Raz within tags as

suggested by Lemay, because Lemay teaches the saving of trouble, and receiving immediate

feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save

time and trouble by inserting the Javascript validation functions in the created HTML tags—

custom tags.

Regarding claim 32, the Applicants state that Raz does not teach or suggest the receipt of

a form definition including one or more custom tags (page 13). The Examiner disagrees, because

Raz teaches the automatic conversion of paper forms into HTML-coded forms—*form*

definition(col.12, lines 36-49).

Moreover, Raz teaches that after converting the forms, validation functions are added to the HTML form fields for verifying the data input into those fields — *identifying and adding validation code that, when executed based on an input corresponding to the* field, *validates whether the associated restrictions are satisfied* (col.12, lines 36-49). In other words, the converted HTML-coded form(s) is received, and then validation functions associated—*one or more associated input restrictions*-- with the HTML form fields, are added to the form. Raz fails to explicitly disclose: *identifying a replacement non-custom tag, adding the identified replacement non-custom tag to a new form definition.* However, Lemay teaches adding validation functions to HTML forms along with tags— *identifying and adding a replacement non-custom tag.* The validation functions are added within HTML tags, and the functions indicate an attribute, such as "ValidLength", and length value not to be exceeded (page 133, lines 17-page 18). It would have been obvious to a person of ordinary skill in the art at the time of the invention to have added the validation tags of Lemay and functions to the form fields of Raz, because Lemay teaches the saving of trouble, and receiving immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a user to save time and trouble by inserting the Javascript validation functions in the created HTML tags—*custom tags.*

Regarding claim 36, the Applicants state that neither Raz nor Lemay teaches or suggests the form and the one or more attributes (page 14). The Examiner disagrees, because

Raz also teaches the automatic conversion of paper forms into HTML forms. During the automatic conversion, validation functions are added to the converted form fields for verifying the data input into those fields (col.12, lines 36-49). Raz fails to explicitly disclose: *each input*

custom tag includes one or more attributes that identify the one or more associated restrictions,

and wherein each of the one or more attributes includes an indication of the attribute and a

corresponding value for that data input corresponding to the tag is to be restricted to. However,

Lemay teaches adding validation functions to HTML forms. The validation functions are added

within HTML tags, and the functions indicate <u>an attribute</u>, such as "ValidLength", and length

value not to be exceeded (page 133, lines 17-page 18). It would have been obvious to a person of

ordinary skill in the art at the time of the invention to include the validation functions of Raz

within attributes of Lemay, because Lemay teaches the saving of trouble, and receiving

immediate feedback without having to wait on a server (page 132, lines 7-10). Thus, allowing a

user to save time and trouble by inserting the Javascript validation functions in the created

HTML tags.

Regarding claim 38, the Applicants submit that " It appears from this rejection that the

HTML form fields of Raz are relied on as disclosing the input field for a form of claim 38.

However. if this were the case, then in order to satisfy the language of claim 38 Raz would need

to disclose a data structure that includes a first portion identifying those HTML form fields and a

second portion identifying one or more restrictions on inputs to the HTML form fields and

further identifying validation code to be added to a page to enforce the one or more restrictions

on inputs to the HTML form fields. Applicant respectfully submits that there is no such data

structure disclosed or discussed in Raz.. Although Raz states that validation functions can be

added to each field to do basic data checking and validation at le client (see, col. 12, lines 44-

45)". The Examiner disagrees, because Raz teaches the adding of validation function having

certain restrictions –*second field with validation code to enforce one or more restrictions*--to be

applied to HTML form fields—*first portion identifying an input field* (col.12, lines 36-49).

Claims 25, 39-42 are rejected at least based on the statements above.

Conclusion

14.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing

date of this final action.

I.   Any inquiry concerning this communication or earlier communications from the examiner

should be directed to Cesar B. Paula whose telephone number is (571) 272-4128. The examiner

can normally be reached on Monday through Friday from 8:00 a.m. to 4:00 p.m. (EST).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Stephen Hong, can be reached on (571) 272-4124. However, in such a case, please

allow at least one business day.

Information regarding the status of an application may be obtained from the Patent

Application Retrieval (PAIR) system. Status information for published applications may be

obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, go to http://portal.uspto.gov/external/portal/pair. Should you have any questions about

access to the Private PAIR system, please contact the Electronic Business Center (EBC) at 866

217-9197 (toll-free).

   Any response to this Action should be mailed to:
         Commissioner for Patents
          P.O. Box 1450
      Alexandria, VA 22313-1450
   Or faxed to:
- **(571)-273-8300** (for **all** Formal communications intended for entry)

CESAR PAULA
PRIMARY EXAMINER
8/4/05